This ICCV workshop paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the accepted version; the final published version of the proceedings is available on IEEE Xplore.

Towards Automated Regulation of *Jacobaea Vulgaris* in Grassland using Deep Neural Networks

Moritz Schauer Renke Hohl Dennis Vaupel Diethelm Bienhaus Seyed Eghbal Ghobadi

University of Applied Sciences Mittelhessen

{first_name}.{second_name}@mni.thm.de

Abstract

The highly poisonous ragwort (Jacobaea Vulgaris) is increasingly spreading, posing significant risks to agriculture, livestock, and nature conservation due to the production of toxic pyrrolizidine alkaloids (PAs). The current manual control methods, such as plucking weed, are labor-intensive and time-consuming. This paper introduces a workflow towards automated regulation of J. Vulgaris, which consists of the two independent tasks of deep learning-based monitoring and controlling. We aim to detect and control J. Vulgaris in an early growth stage before the plant can reseed, which challenges the data collection and the training of deep neural networks. Primarily we need to detect the green leaf rosettes on a green meadow. The main focus lies on the monitoring part with synthetic training data generation and a deep neural network-based labeling assistant.

1. Introduction

Since the middle of the last decade, an increased spread of ragwort (*Jacobaea Vulgaris* Gaertn., syn. *Senecio jacobaea* L.) has been observed in Germany [10, 22]. The plant spreads mainly in nature protected areas and extensively used grassland. Therefore, meadows which are used for animal feed production are affected. This is especially problematic because ragwort produces the poisonous pyrrolizidine alkaloid (PA), which is toxic to mammals. Canned feed, such as hay or silage, may contain the toxin, leading to potential poisoning incidents in livestock. Due to the potential danger of poisoning, ragwort mass occurrences endanger the usability and acceptance of species-rich, extensively used grassland stands that are valuable for nature conservation. In order to use the infested areas after all, regular control and eradication of ragwort is necessary. Currently, this has been a manual and time-consuming process. Manual control is usually performed shortly before mowing. At that point, the plants have already formed yellow flowers and are easily recognizable. A significant disadvantage of this approach is that the plants can reseed if not all

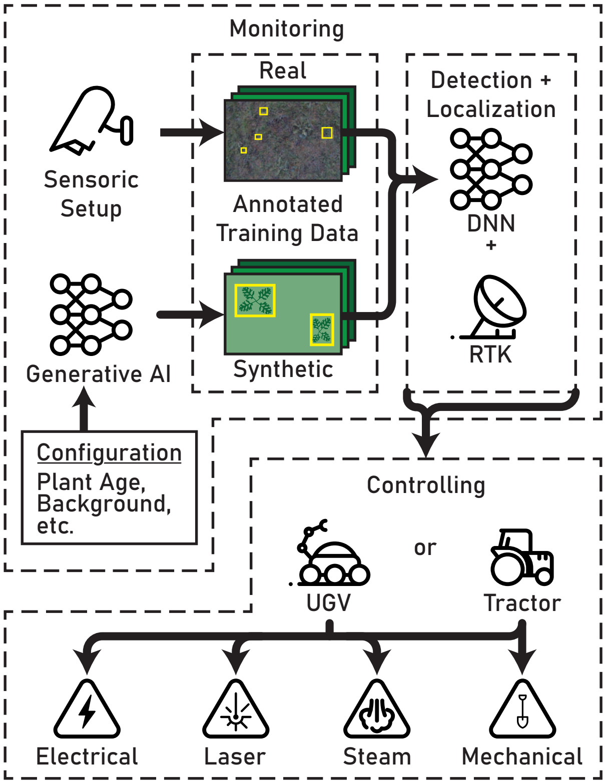


Figure 1: Proposed concept overview of *J. Vulgaris* regulation

This work's writing was partly enabled from within the context of the project "KIhUG - AI assisted highly automated weed control in grasslands", funded by the strategic research fund of the University of Applied Sciences Mittelhessen.

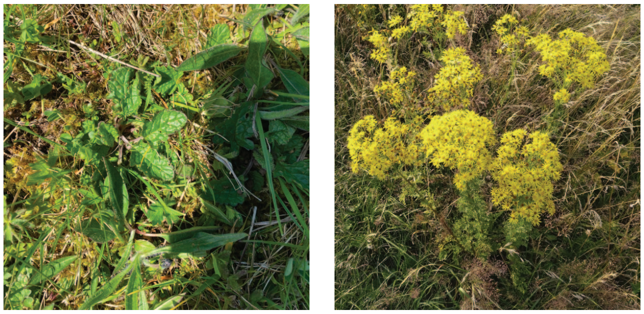


Figure 2: *J. Vulgaris* rosette in first year growth stage (a) and in the second year growth stage with erect stems with inflorescences (b).

plants are removed before the seeds are ripe. Individual *J. Vulgaris* plants have the capacity to produce up to 50000 seeds, and in certain cases, even more [13]. New *J. Vulgaris* plants can thus sprout from the soil seed bank even after several years. Therefore, the approach proposed in this paper, depicted in Fig. 1, takes advantage of the fact that *J. Vulgaris* are monocarpic biennial plants. In the first year of their two-year growth cycle, they form a basal rosette of leaves, and only in the second year they grow an erect stem with inflorescences, as shown in Fig. 2. After it has flowered, the plant dies. Detection and control of *J. Vulgaris* in the first year of the growth phase prevents the plant from reseeding. New challenges arise, as the plant can no longer be identified solely by the easily recognizable yellow flowers but should be determined based on the formed leaf rosette. While there are already many efforts and publications in the field of weed identification, they mainly focus on detecting weeds in arable crops with easy soil backgrounds. In grassland cultivation, however, the green leaf rosettes on a green meadow should be detected. Furthermore, the diversity of species in these meadows and the resulting frequency of plants that look similar to *J. Vulgaris* (e.g., *Ajuga Reptans*) is challenging.

2. Related Work

Unmanned ground vehicles (UGVs) in agriculture are a widely researched topic. However, most studies focus on usage in arable farming, especially in the field of weed detection and control. In classical arable farming, methods for weed detection that rely on the spectral properties of plants are particularly promising [21, 28, 25]. The implementation of these methods for managing grasslands has been constrained due to the diversity of vegetation. The authors in [11, 36, 38] have shown promising results based on deep neural networks and using RGB images. However, approaches based on Deep Neural Networks (DNNs) usually require a large amount of data. This is still a limiting factor for employing DNNs in many areas for which no data is available. Without having public datasets, the data needs to be generated. This process includes data recording and annotation. Especially the annotation part is very time-consuming and cumbersome. Recent research shows that using neural networks can support the annotation process or even automate it completely [27]. Schilling *et al.* [35] presented a tool that enables assisted annotation in image processing tasks, simplifying the process, increasing efficiency, improving annotation quality, and offering additional functionalities to support users. Another approach for reducing annotation efforts is proposed by Rettneberger *et al.* [31]. They use a thresholding technique to get simple masks on which a neural network is trained. However, the methods presented are mainly focused on medical images and cannot be easily transferred to grassland images. Another way to get annotated data is to generate the data synthetically. In this case, the annotations can be generated directly. In addition, synthetic data has the advantage of eliminating the time-consuming process of collecting real data. The use of synthetic training data in training neural networks is rapidly growing. Vietz *et al.* [37] have shown that artificially generated data can fill up potential gaps in datasets. Also, in the field of weed identification, the use of model-based synthetic data has been evaluated by Iqbal *et al.* [19]. Moreover, the generation of synthetic data finds applications beyond 2D image processing. For example, Chaudhury *et al.* [4] have shown that the segmentation of point clouds of real plants can also be trained with the help of generated 3D models. However, the application of generative AI models to create synthetic grassland datasets is novel and unexplored. The rise of generative AI models has opened up new possibilities for synthetic data generation and augmentation. For instance, Antoniou *et al.* [1] established that the inclusion of a DAGAN in the workflow can enhance the performance of classifiers, even after the application of conventional data augmentation methods. Furthermore, He *et al.* [15] found that state-of-the-art text-to-image diffusion models are well fit to provide training data for recognition tasks, especially in zero-shot or few-shot scenarios. Fine-tuning of large pre-trained diffusion models for synthetic image dataset creation has mainly been applied to medical and microbiological datasets[3, 26].

3. Proposed Workflow

Our proposed approach is divided into the two tasks of monitoring and controlling, as illustrated in Fig. 1. Both tasks are performed in independent steps. For example, monitoring could be carried out with a drone or camera system attached to a tractor. The same applies to the control of the plants, where various approaches are conceivable, ranging from a tractor with attached tools to a UGV that inde-

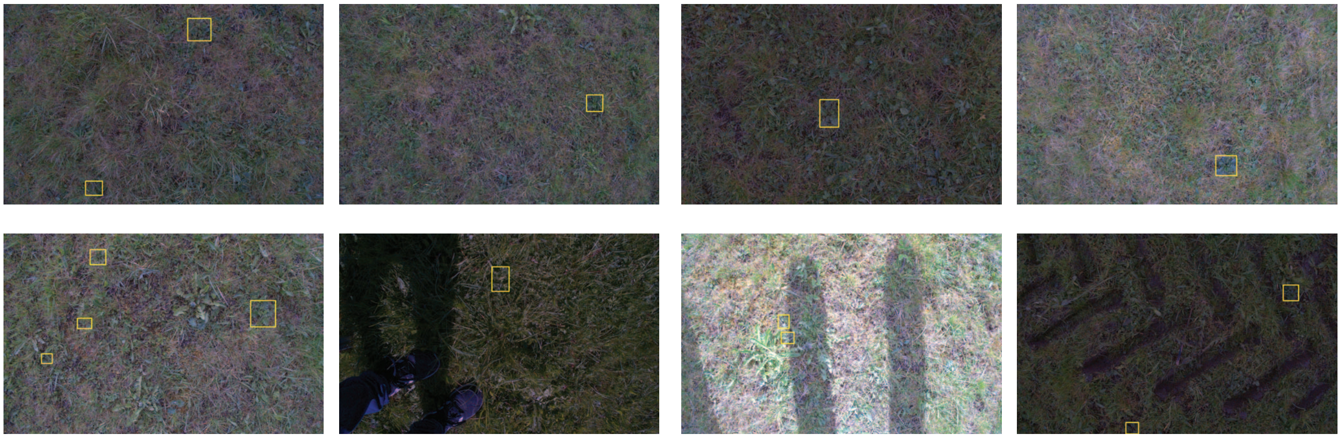


Figure 3: Different samples from the acquired dataset. The top row represents relatively easy samples where the *J. Vulgaris* plants are easier recognizable. More complex examples are shown on the bottom row, with smaller plants, interference objects, shadows, or even tractor tracks.

pendently drives the field based on the previously collected data. In the following, we will explain the two stages in more detail, as well as the individual approaches.

3.1. Monitoring

Sensoric Setup During monitoring and controlling, we need to know the exact position of the plants. For this purpose, we use at least one camera mounted on a carrier platform. The carrier platform can be a tractor, a drone, or a UGV. In the first place, this camera is a typical industrial RGB camera. The setup for creating the first dataset will be described in more detail in section 4.1.

Data Collection and Annotation Extensively annotated datasets are essential to train efficient DNNs that recognize the plants in the meadow. So the first step in the monitoring process is to collect the data. Therefore, we use our proposed sensor setup to capture images of the meadow. Additionally, we propose to use synthetic data to extend the real dataset. In the annotation process, synthetic data have the advantage of allowing direct annotation generation. Real data, on the other hand, are annotated using a labeling tool. Since this process is time-consuming and only some plants are easily recognizable, we aim to support this process with a weak classifier. In the first place, we only annotate the *J. Vulgaris* plants with bounding boxes.

Detection and Localization After collecting and annotating the data, we employ DNNs to detect and localize *J. Vulgaris* plants based on object detection and image segmentation techniques. Both approaches can be trained to detect whole plants or single leaves in an image. We utilize a combination of real and synthetic data to train the networks. Subsequently, the detected plant locations in the images are combined with position data from the carrier platform to determine the plants' locations in the meadow precisely. Upon successful detection and localization, we obtain a map of the meadow with the exact positions of the *J. Vulgaris* plants. The map serves as a valuable resource for understanding the distribution and abundance of *J. Vulgaris* within the meadow, aiding in the formulation of effective control and management strategies.

3.2. Controlling

Besides the perception part, different controlling strategies for the detected plants should be developed and evaluated. Currently, the regulation of *J. Vulgaris* is usually performed manually. Other approaches have been investigated, such as regulation by varied cutting and restoration measures [40]. However, the disadvantage of these approaches is that they are either very time-consuming or cannot target individual plants. Our approach generates a map during the monitoring phase to give the farmer a suitable tool for controlling the plants. Based on this map, the farmer can decide on a controlling strategy. This could still be manual control if the number of plants is low or a targeted control with an automated approach if the number of plants is high. With the assumption that we detect many plants, we aim at a targeted control of individual plants. Here the main focus will be on methods such as mechanical control [24], hot air [7], laser weeding [39], or treatment by applying high voltage [6]. Aside from the success rate, the processing speed plays a crucial role in determining the appropriate control methods. The choice of control approach is influenced by the speed at which grassland processing occurs, typically up to $12\,\mathrm{km\,h^{-1}}$. However, this limitation can be addressed by employing UGVs as their speed can be adjusted accordingly.

4. Methods and Implementation

In the previous section, we outlined our roadmap for controlling *J. Vulgaris* in our research project. This includes key components such as data generation and annotation, crucial for effective *J. Vulgaris* control. In this section, we will focus on detailing the implementation of these techniques. It is important to note that the rest of the proposed workflow, including other components such as segmentation and control strategies, will be shown in our forthcoming research project publications.

4.1. Dataset

As a first approach, a dataset was recorded on two meadows in Zehnhausen near Rennerod in Rhineland-Palatinate, Germany. A Basler ace 2 camera with a global shutter and an image resolution of $1920 \times 1200\,\mathrm{px}$ was used for the recordings. The camera was equipped with a Basler lens with a focal length of 4 mm. During the recording, different recording strategies were tested. First, the camera was attached to the pitchfork of a tractor. Then the tractor was driven over the meadow at three typical processing speeds (3, 5, and 12 km/h). Not the whole meadow was covered, but only a part of it. For the second approach, the camera was attached to a portable frame. With the help of the frame, specific pictures were taken in which *J. Vulgaris* is visible. Additionally, several close-ups of single plants were recorded using a 12 MP camera. Next, we annotated the images using the labeling tool CVAT[1]. As labels, we use bounding boxes in the YOLO format [30] to train object detection algorithms. However, a pixel-wise annotation for segmentation is also conceivable. Each bounding box includes the whole plant with all attached leaves. The assignment choice for the *J. Vulgaris* class relied solely on the visual characteristics depicted in the images without considering any real-world plant phenotype. An excerpt from the dataset is shown in Fig. 3. Here, one can see the heterogeneity of the dataset, for example, the different lighting conditions or the number of *J. Vulgaris* plants per image.

4.2. Annotation using Weak Classifier

The primary goal of the weak classifier is to train a neural network with a small amount of data, which will support and simplify the labeling of further data. In addition, initial tests can determine whether a neural network can recognize the characteristics of *J. Vulgaris* and thus distinguish the plant from others. Therefore, we introduce a weak classifier called *Easy Label Assist Net (ELAN)*, which generates a heatmap, as illustrated in Fig. 4. This heatmap indicates where potentially *J. Vulgaris* plants are located in a new image. To generate the heatmap, we divide a given image into $N \times N$ patches and process each patch in a multi-stage process. First, we train a feature extractor to recognize features in the patches. Next, we build and train a classifier on top of this feature extractor to predict whether an image patch contains *J. Vulgaris*. Finally, we stitch the patches together and upscale the heatmap to match the original resolution.

4.2.1 Training data

The previously acquired dataset (see Sec. 4.1) was created for object detection applications. In order to use this dataset for classification tasks, the labeled bounding boxes on which *J. Vulgaris* is seen are cropped from the images with a size of $128 \times 128\,\mathrm{px}$. Counterexamples were generated by randomly selecting image crops of the same size that do not overlap with *J. Vulgaris* bounding boxes. To perform the network's training, we created a well-balanced dataset comprising a total of 500 samples. In addition, we created a dataset of 500 synthetic images with our proposed data generation (see Sec. 4.3.2).

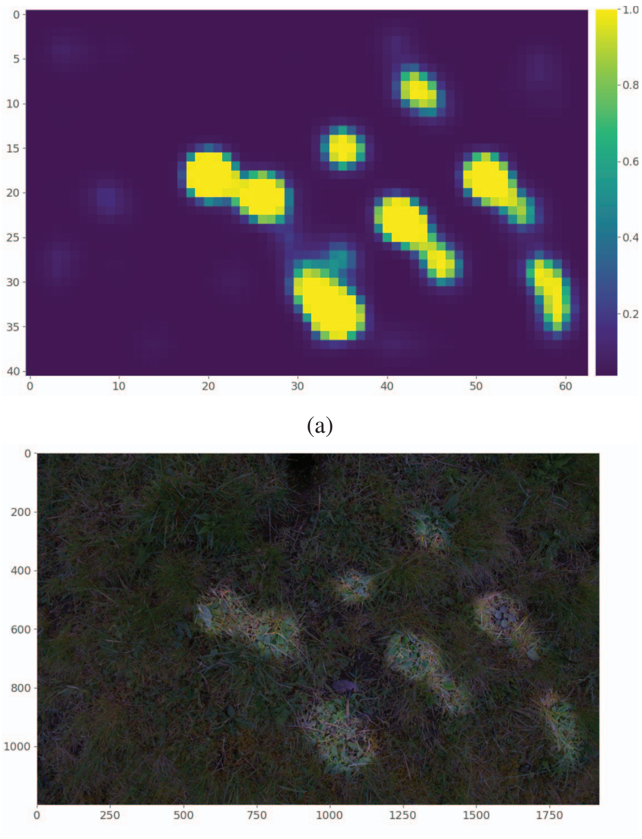
4.2.2 Training of the feature extractor

Since the amount of training data is small, we applied a few-shot learning strategy, presented in Hadsell *et al.* [12]. The proposed architecture, called *Siamese Neural Network (SNN)*, can be seen as the learnable distance metric $D_W(\vec{X}_1, \vec{X}_2) = ||G_W(\vec{X}_1) - G_W(\vec{X}_2)||_2$ between two images $\vec{X}_1$ and $\vec{X}_2$. Here, G_W denotes an underlying CNN model that outputs embeddings of the corresponding images, between which the Euclidean distance is measured. The objective is to decrease the distance D_W between images of the same category while increasing the distance between images of different categories. To this end, we minimize the so-called contrastive loss function:

$$L(W, Y, \vec{X}_1, \vec{X}_2) = \\ (1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{\max(0, m - D_W)\}^2 \quad (1)$$

where $Y \in \{0, 1\}$ describes whether $\vec{X}_1$ and $\vec{X}_2$ belong to the same or different categories. Each of the summands covers a specific case. If $\vec{X}_1$ and $\vec{X}_2$ belong to the same category, we want to minimize $\frac{1}{2}(D_W)^2$. In the case that $\vec{X}_1$ and $\vec{X}_2$ belong to different categories, we want to minimize $\frac{1}{2}\{\max(0, m - D_W)\}^2$. The upper margin $m > 0$ is necessary to prevent the network from lowering the loss by raising the distance of different instances to infinity. It defines the radius in which the distance of dissimilar pairs contributes to the loss. We assigned the value of m as 20. The SNN uses an underlying CNN model where parameters can be optimized to learn the correct distances. While the

[1] https://www.cvat.ai/



(a)

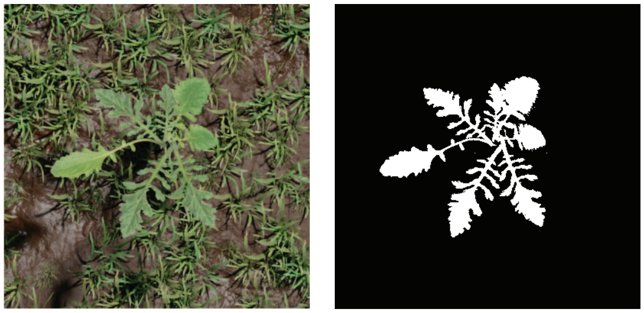
(b)

Figure 4: (a) Example of a generated heatmap where each pixel shows the confidence whether *J. Vulgaris* is included in the image patch or not. (b) Original image from the dataset, over which the generated heatmap was overlayed.

architecture of the CNN model G_W can be freely chosen, a pre-trained ResNet50 [14] worked best in our case. For the training process, pairs are formed from all dataset images, which will be used for the training. Thereby, the number of training samples is increased by a factor of $0.5(n-1)n$, where n is the number of used samples. For example, we generated a training dataset of 124750 image pairs from the 500 underlying training images.

4.2.3 Training of the PatchGAN classifier

After training the feature extractor G_W, we used it to build a classifier based on the PatchGAN approach [20]. The PathcGAN is a pure convolutional model where the last convolutional layer outputs one feature map containing the classifications of each patch. The peculiarity of the architecture is that it can process images of arbitrary size. Therefore, the network's output is a heatmap with the same aspect ratio as the input image, each pixel indicating whether *J. Vulgaris* is observable in the area or not. The size of the heatmap is determined by the number N of image patches in an image. In our case, we rebuild network architecture to classify image patches of size $128 \times 128\,\mathrm{px}$. From this follows that the training images with a size of $128 \times 128\,\mathrm{px}$ result in $1 \times 1\,\mathrm{px}$ heatmaps. For the whole images in our dataset with a size of $1920 \times 1200\,\mathrm{px}$, the resulting heatmaps are $57 \times 35\,\mathrm{px}$. Since the feature extractor is already trained, it is sufficient to train only the classification layer. The objective of the classifier is binary classification, distinguishing between *J. Vulgaris* and no*J. Vulgaris*kk patches. We employed the Binary Cross Entropy loss function, effectively training the classifier to identify the presence of *J. Vulgaris* in image patches.



(a)

(b)

Figure 5: Simulated 3D environment with a custom *J. Vulgaris* model (a) and the generated segmentation mask (b).

4.3. Synthetic Training Data

Capturing labeled data of *J. Vulgaris* poses significant challenges due to its time-consuming nature and restrictions on data collection during specific periods of the year. As a result, the acquired dataset is relatively small and likely exhibits an imbalance across lighting conditions, growth stages, and terrains. This section explores various approaches for generating annotated images of *J. Vulgaris* to augment our training dataset and improve detection performance. Firstly, we will briefly describe a model-based approach, which offers promising possibilities for generating additional data. Subsequently, we will describe a more elaborate approach using generative models to enhance the dataset further.

4.3.1 Model-based approach

With this approach, we generate 3D models of *J. Vulgaris* within its natural surroundings and transform these scenes into visual representations. The key advantage of this methodology lies in the precise manipulation of the rendering procedure. It allows us to flexibly simulate various lighting conditions, perspectives, camera configurations, and the spatial arrangement of objects within the scene. To implement the 3D rendering process, we leveraged the capabilities of the open-source 3D modeling tool Blender[2]. To create realistic representations of *J. Vulgaris*, we utilized custom meshes and textures specific t*J. Vulgaris*kk. Additionally, we incorporated various environmental assets such as grass, weeds, and other disruptive objects. To introduce diversity and variation, we introduced randomization in factors like leaf arrangement, plant positioning, and the number of *J. Vulgaris* instances within a scene. Our custom rendering process also allows for the generation of segmentation masks (Fig. 5). These pixel-perfect annotations could be used for image segmentation training in the future. Additionally, we could output the bounding boxes easily as we know exactly where we placed the *J. Vulgaris* plants. This approach, however, has some crucial shortcomings. First, 3D modeling requires expert knowledge of *J. Vulgaris*'s characteristics to model ”ideal” specimens. It is also difficult to model plant variance since it is difficult to identify from what point a plant is no longer recognizable as a specimen of *J. Vulgaris*. A more fundamental problem is the obvious existence of a reality gap: Rendered images can easily be identified as computer-generated imagery. Much effort can be spent on shrinking that gap, but true photorealism is almost impossible to realize. The next section will explain a more promising approach using generative AI.

4.3.2 Neural network-based approach

Our second synthetic data-generating approach is leveraging state-of-the-art generative AI to create artificial images. Until recently, Generative Adversarial Networks (GANs) have been the most popular architecture for generating images. They have also been used in various ways for precision agriculture and weed control [8, 23, 9]. In recent years, there has been an upsurge in diffusion-based architectures, with models like DALL·E [29], Imagen [34] and Stable Diffusion (SD) [32] being the most prominent implementations. It has been shown that diffusion-based models can provide better image quality than GANs [5]. They also don't suffer from mode collapse, a common problem during GAN training. In the case of diffusion, a significant drawback is a noticeable increase in the inference time [2].

Model selection Our approach utilizes Stable Diffusion [32], a text-to-image model specifically designed for high-resolution image synthesis. Stable Diffusion leverages the basic diffusion process described in [17]. It employs a denoising network that progressively constructs images from Gaussian noise, operating on latent image representations to expedite training and inference. SD's image generation is conditioned by text prompts, allowing for the deliberate synthesis of *J. Vulgaris*'s growth stages and terrains, provided they are in sufficient quantities in the training data. In addition to its primary text-to-image pipeline, SD can perform various downstream tasks, including inpainting, outpainting, image-to-image translation, and image upscaling. We propose using text-conditioned inpainting to insert *J. Vulgaris* onto real backgrounds at specific positions within given inpainting masks (refer to Fig. 6). While the positions of generated objects may not be pixel-perfect, the precision of bounding boxes derived from inpainting masks has been found to be comparable to manually created bounding boxes.

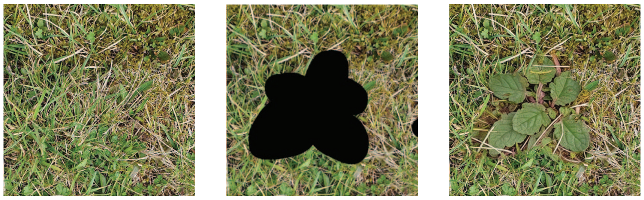


Figure 6: On a real background (left), limited by an inpaint mask (center), a synthetic weed is inserted (right).

Model Fine-tuning Since the images we want to generate are very different from typical training datasets, we need to fit the model to our data. Therefore, we need to create an image-text dataset from our data. To create such dataset we can utilize our existing object detection dataset. We can use the bounding boxes to crop the objects of interest and caption them with certain ”trigger words” which can later be used in prompts. For the fine-tuning process itself one can choose various techniques. The most used variants are full model fine-tuning, Dreambooth [33], and Low-rank adaptiont (LoRA) [18]. Dreambooth and LoRA are specifically designed for extreme few-shot learning, utilizing as few as 10 to 20 training images. These approaches work because the base model has already been trained on many related objects. A complete fine-tuning of the model is necessary to adapt to our images properly, as our training images differ significantly from those used to train the base model.

5. Experiments and Results

This section presents the experiments and results we carried out for our proposed methods, starting with the evaluation of the *ELAN* model and followed by the synthetic image generation.

5.1. Annotation using Weak Classifier

We have trained three versions of the *ELAN* model. $ELAN_{Real}$ was trained with real data only, $ELAN_{Synth}$ with synthetic data only, and $ELAN_{Mix}$ with mixed data consisting of equal parts of real and synthetic data. All

[2] https://www.blender.org/

Subset	Model	Accuracy	Precision	Recall	F1	Average Precision
Val_{Real}	$ELAN_{Real}$	0.9540	0.9619	0.9385	0.9500	0.9314
	$ELAN_{Synth}$	0.8793	0.9466	0.7854	0.8585	0.8434
	$ELAN_{Mix}$	0.9187	0.8752	0.9628	0.9169	0.8599
Val_{Synth}	$ELAN_{Real}$	0.9932	1.0000	0.9875	0.9937	0.9943

Table 1: Evaluation results of the *ELAN* classifiers on validation dataset as well as the synthetic dataset.

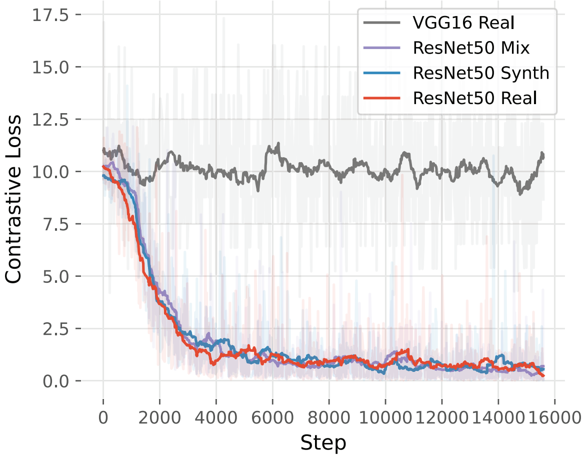


Figure 7: Contrastive loss of training multiple feature extractor models on real, synthetic, and mixed data over two epochs (7797 steps per epoch). The VGG16 was only trained on real data.

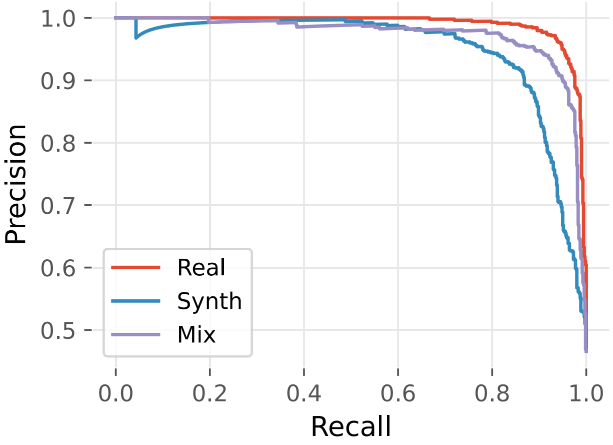


Figure 8: Precision-Recall curves of various *ELAN* models, evaluated on the validation dataset.

training datasets had a size of 500 images each. For each model, a feature extractor had to be trained first, followed by the training of the corresponding PatchGAN classifier.

Hyperparameters We trained each feature extractor over 2 epochs (7797 steps per epoch) with a batch size of 16, which means that 16 image pairs result in a total of 32 images per batch. The PatchGAN classifiers were trained with a batch size of 32 over 10 epochs. While training a PatchGAN classifier, we no longer adjusted the feature extractor's weights. An Adam optimizer with a learning rate of 1×10^{-4} was chosen for both the feature extractors and the PatchGAN classifiers. In addition, we applied several augmentation techniques during the whole training, including random horizontal and vertical flip, rotation, translation, scale, shear, grayscale, and brightness adjustment.

Results Since generating heatmaps is a classification task for individual image patches, we evaluated each *ELAN* model by measuring classification metrics on a validation dataset consisting of 1500 real samples. In addition, $ELAN_{Real}$ was evaluated on the synthetic dataset consisting of 500 synthetic samples. Initially, we tested two different backbones for our feature extractor. However, the results showed that, in contrast to the ResNet50 backbone, the VGG16 backbone could not learn the data features, which can be seen in Fig. 7. Therefore, all further experiments were only done with the ResNet50 architecture. The evaluation results of the different *ELAN* models can be seen in table 1. The precision-recall-curves in Fig. 8 show that all versions of *ELAN* are capable of recognizing *J. Vulgaris*. As described earlier, the resulting heatmaps are significantly smaller than the original images. In order to use them as a labeling aid, they must be scaled up to the original size in the final step. An example of an image prelabeled by $ELAN_{Real}$ is shown in Fig. 4b.

5.2. Synthetic Training Data

For the full model fine-tuning, we trained a SD v1.5 model using EveryDream Trainer 2.0 [3], which provides advanced functionalities for model training and optimization. As a training dataset, we used 110 images of *J. Vulgaris* and background at a resolution of $256 \times 256\,\mathrm{px}$.

Hyperparameters During training, we employed a batch size of 6 and maintained a constant learning rate of 5×10^{-7} for both the text encoder and U-Net. The model was trained for 100 epochs. Through qualitative assessment of the model checkpoints, it became noticeable after 100 epochs that the variance of the samples decreased significantly, i.e., more similar plants were generated. The training images were cropped to squares and captioned with descriptions containing the trigger word *jacobaea*. It is worth noting that the training script used in this experiment did not support lower resolutions, necessitating the upscaling of the images to $256 \times 256\,\mathrm{px}$.

Results Using the proposed training procedure, the resulting model can produce images that closely resemble the training images regarding weed shapes, lighting conditions, and overall visual appearance. As depicted in Fig. 9, the model can generate fairly realistic images of a wide variety. This general qualitative assessment is supported by the Fréchet Inception Distance (FID) [16] of 55.5 calculated over 110 samples of real and synthetic images each. Here, smaller values represent better image quality. In evaluating the performance of our *ELAN* classifier, we found that the model trained on a dataset consisting only of real images performs equally well on synthetically generated images (Table 1). This result suggests that while the synthetic images already capture the different characteristics of *J. Vulgaris* and its environment to some degree, they still need more crucial details to appropriately train neural networks on them. Therefore, to enhance the quality and effectiveness of the synthetic data, we plan to train our SD model on higher-resolution images and a larger dataset. Currently, a significant portion of the available training images falls below the minimum resolution requirement of $256 \times 256\,\mathrm{px}$. Consequently, upscaling has introduced distortions that degrade the quality of the synthetic images. Addressing this resolution discrepancy will be a key focus in future iterations of the model training process.

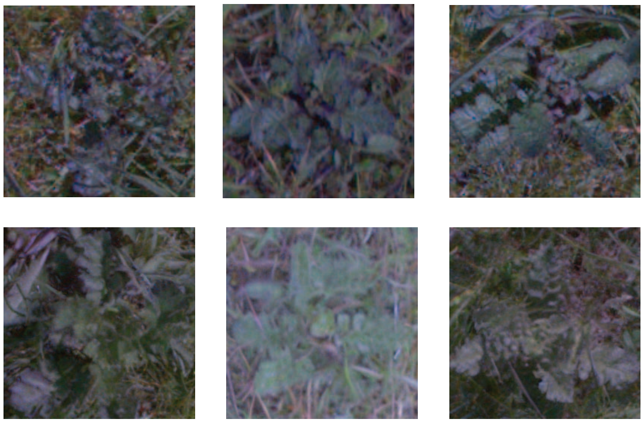


Figure 9: Real training samples (top row) and samples generated by our finetuned model (bottom row).

6. Conclusion and Future Work

Our work proposes a comprehensive approach to effectively regulate the poisonous plant *J. Vulgaris*. One of the key advantages of our modular approach lies in the two independent tasks, enabling us to fully exploit the potential of appropriate carrier platforms and controlling strategies. In addition to the proposed conceptual workflow, we presented our work on synthetic training data generation and a labeling aid. Initial results indicated that annotation time can be reduced using our *ELAN*. Leveraging the Siamese architecture, we have successfully trained a classifier with only a limited amount of data. While our work highlights the potential of synthetic data generation for grassland robotics, we acknowledge several open challenges. The need for more diverse data for object detection algorithms and higher-resolution images for image generation is crucial. We have also explored the potential of SD-based image generation, but further experiments are required to understand its capabilities and limitations fully. We plan to explore using time-dependent image generation for synthetic training data in future investigations. Furthermore, we are investigating various object detection architectures that exhibit promising results. While focusing on the monitoring aspect of our proposed workflow, our future work will also examine different control methods. In this context, we will focus on integrating the overall system with attention to processing speed, quality, and reliability. Accurately monitoring plants using position data will be a crucial development area for effectively controlling *J. Vulgaris*. By addressing these challenges and pursuing future research directions, we aim to advance the state-of-the-art in regulating *J. Vulgaris*, ultimately leading to improved control strategies with enhanced efficiency and effectiveness.

References

[1] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Augmenting image classifiers using data augmentation generative adversarial networks. In *2018 Artificial Neural Networks and Machine Learning (ICANN)*, volume 11141,

[3] https://github.com/victorchall/EveryDream2trainer

pages 594–603, Cham, 2018. Springer International Publishing. 2

[2] Sam Bond-Taylor, Adam Leach, Yang Long, and Chris G. Willcocks. Deep Generative Modelling: A Comparative Review of VAEs, GANs, Normalizing Flows, Energy-Based and Autoregressive Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7327–7347, Nov. 2022. arXiv:2103.04922 [cs, stat]. 6

[3] Pierre Chambon, Christian Bluethgen, Jean-Benoit Delbrouck, Rogier Van der Sluijs, Małgorzata Połacin, Juan Manuel Zambrano Chaves, Tanishq Mathew Abraham, Shivanshu Purohit, Curtis P. Langlotz, and Akshay Chaudhari. RoentGen: Vision-Language Foundation Model for Chest X-ray Generation, Nov. 2022. arXiv:2211.12737 [cs]. 2

[4] Ayan Chaudhury, Peter Hanappe, Romain Azaïs, Christophe Godin, and David Colliaux. Transferring PointNet++ Segmentation from Virtual to Real Plants. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, Montreal, Canada, Oct. 2021. 2

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *2021 Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 8780–8794. Curran Associates, Inc., 2021. 6

[6] M. F. Diprose and F. A. Benson. Electrical methods of killing plants. *Journal of Agricultural Engineering Research*, 30:197–209, Jan. 1984. 3

[7] Ahmed Shawky El_Sayed and AbdelGawad Saad. Development of a Field Weed Control Device Using Hot Air: Weed control using Hot Air. *Agricultural Engineering International: CIGR Journal*, 25(2), June 2023. Number: 2. 3

[8] Borja Espejo-Garcia, Nikos Mylonas, Loukas Athanasakos, Eleanna Vali, and Spyros Fountas. Combining generative adversarial networks and agricultural transfer learning for weeds identification. *Biosystems Engineering*, 204:79–89, Apr. 2021. 6

[9] Mulham Fawakherji, Ciro Potena, Alberto Pretto, Domenico D. Bloisi, and Daniele Nardi. Multi-Spectral Image Synthesis for Crop/Weed Segmentation in Precision Farming. *Robotics and Autonomous Systems*, 146:103861, Dec. 2021. 6

[10] Christoph Gottschalk, Florian Kaltner, Matthias Zimmermann, Rainer Korten, Oliver Morris, Karin Schwaiger, and Manfred Gareis. Spread of Jacobaea vulgaris and Occurrence of Pyrrolizidine Alkaloids in Regionally Produced Honeys from Northern Germany: Inter- and Intra-Site Variations and Risk Assessment for Special Consumer Groups. *Toxins*, 12(7):441, July 2020. 1

[11] Ronja Güldenring, Evangelos Boukas, Ole Ravn, and Lazaros Nalpantidis. Few-leaf Learning: Weed Segmentation in Grasslands. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3248–3254, Prague, Czech Republic, Sept. 2021. IEEE. 2

[12] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1735–1742, New York, NY, USA, 2006. IEEE. 4

[13] John L. Harper and W. A. Wood. Senecio Jacobaea L. *The Journal of Ecology*, 45(2):617, July 1957. 2

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA, June 2016. IEEE. 5

[15] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and XIAOJUAN QI. Is synthetic data from generative models ready for image recognition? In *2023 International Conference on Learning Representations (ICLR)*, 2023. 2

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *2017 International Conference on Neural Information Processing Systems (NeurIPS)*, page 6629–6640, Red Hook, NY, USA, 2017. Curran Associates Inc. 8

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. volume 33, pages 6840–6851, 2020. 6

[18] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *2022 International Conference on Learning Representations (ICLR)*, 2022. 6

[19] Naeem Iqbal, Justus Bracke, Anton Elmiger, Hunaid Hameed, and Kai von Szadkowski. Evaluating synthetic vs. real data generation for ai-based selective weeding. In Christa Hoffmann, Anthony Stein, Arno Ruckelshausen, Henning Müller, Thilo Steckel, and Helga Floto, editors, *43. GIL-Jahrestagung, Resiliente Agri-Food-Systeme*, pages 125–135, Bonn, 2023. Gesellschaft für Informatik e.V. 2

[20] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, Los Alamitos, CA, USA, jul 2017. IEEE Computer Society. 5

[21] Tang Jinglei, Miao Ronghui, Zhang Zhiyong, Xin Jing, and Wang Dong. Distance-based separability criterion of ROI in classification of farmland hyper-spectral images. *International Journal of Agricultural and Biological Engineering*, 10(5):177–185, Sept. 2017. 2

[22] Stefanie Jung, Jan Lauter, Nicole M. Hartung, Anja These, Gerd Hamscher, and Volker Wissemann. Genetic and chemical diversity of the toxic herb Jacobaea vulgaris Gaertn. (syn. Senecio jacobaea L.) in Northern Germany. *Phytochemistry*, 172:112235, Apr. 2020. 1

[23] Shahbaz Khan, Muhammad Tufail, Muhammad Tahir Khan, Zubair Ahmad Khan, Javaid Iqbal, and Mansoor Alam. A novel semi-supervised framework for UAV based crop/weed classification. *PLOS ONE*, 16(5):e0251008, May 2021. Publisher: Public Library of Science. 6

[24] Chris McCool, James Beattie, Jennifer Firn, Chris Lehnert, Jason Kulk, Owen Bawden, Raymond Russell, and

Tristan Perez. Efficacy of Mechanical Weeding Tools: A Study Into Alternative Weed Management Strategies Enabled by Robotics. *IEEE Robotics and Automation Letters*, 3(2):1184–1190, Apr. 2018. 3

[25] S. Imran Moazzam, Umar S. Khan, Waqar S. Qureshi, Mohsin I. Tiwana, Nasir Rashid, Waleed S. Alasmary, Javaid Iqbal, and Amir Hamza. A patch-image based classification approach for detection of weeds in sugar beet crop. *IEEE Access*, 9:121698–121715, 2021. 2

[26] Puria Azadi Moghadam, Sanne Van Dalen, Karina C. Martin, Jochen Lennerz, Stephen Yip, Hossein Farahani, and Ali Bashashati. A Morphology Focused Diffusion Probabilistic Model for Synthesis of Histopathology Images. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1999–2008, Waikoloa, HI, USA, Jan. 2023. IEEE. 2

[27] Victor-Emil Neagoe and Radu-Mihai Stoica. A new neural network-based approach for automatic annotation of remote sensing imagery. In *2014 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1781–1784, July 2014. 2

[28] Luan P Pott, Telmo JC Amado, Raí A Schwalbert, Elodio Sebem, Mithila Jugulam, and Ignacio A Ciampitti. Pre-planting weed detection based on ground field spectral data. *Pest Management Science*, 76(3):1173–1182, 2020. 2

[29] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *2021 International Conference on Machine Learning (ICML)*, pages 8821–8831. PMLR, 2021. 6

[30] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, Las Vegas, NV, USA, June 2016. IEEE. 4

[31] Luca Rettenberger, Marcel Schilling, and Markus Reischl. Annotation Efforts in Image Segmentation can be Reduced by Neural Network Bootstrapping. *Current Directions in Biomedical Engineering*, 8(2):329–332, Aug. 2022. 2

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, New Orleans, LA, USA, 2022. 6

[33] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22500–22510, 2023. 6

[34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. volume 35, pages 36479–36494, 2022. 6

[35] Marcel P. Schilling, Svenja Schmelzer, Lukas Klinger, and Markus Reischl. KaIDA: A modular tool for assisting image annotation in deep learning. *Journal of Integrative Bioinformatics*, 19(4), Dec. 2022. 2

[36] Frits K. van Evert, Joost Samsom, Gerrit Polder, Marcel Vijn, Hendrik-Jan van Dooren, Arjan Lamaker, Gerie W.A.M. van der Heijden, Corné Kempenaar, Ton van der Zalm, and Lambertus A.P. Lotz. A robot to detect and control broad-leaved dock (Rumex obtusifolius L.) in grassland. *Journal of Field Robotics*, 28(2):264–277, 2011. 2

[37] Hannes Vietz, Tristan Rauch, and Michael Weyrich. Synthetic training data generation for convolutional neural networks in vision applications. In *2022 IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, Sept. 2022. 2

[38] Jinjin Wang, Xiaopeng Yao, and Bao Kha Nguyen. Identification and localisation of multiple weeds in grassland for removal operation. In *2022 International Conference on Digital Image Processing (ICDIP)*, volume 12342, pages 290–299. SPIE, Oct. 2022. 2

[39] Mingfeng Wang, José-Alfredo Leal-Naranjo, Marco Ceccarelli, and Simon Blackmore. A Novel Two-Degree-of-Freedom Gimbal for Dynamic Laser Weeding: Design, Analysis, and Experimentation. *IEEE/ASME Transactions on Mechatronics*, 27(6):5016–5026, Dec. 2022. Conference Name: IEEE/ASME Transactions on Mechatronics. 3

[40] Henrike Wiggering, Tim Diekötter, and Tobias W. Donath. Regulation of Jacobaea vulgaris by varied cutting and restoration measures. *PLOS ONE*, 17(10):e0248094, Oct. 2022. 3